Spencer Ream
CSIS-2430
Professor Jon McGowan

Monopoly Simulation Report

Summary

The simulation of Monopoly was coded in Python. A system of Object Oriented Programming was used to implement classes and functions that mimic best the board used in Monopoly. It simulates the moves and keeps track of the number of moves made, as well as where the player landed. It also implements Get Out of Jail Free Cards, which are one of the options in both the Community Chest and Chance decks of cards. Although not necessary for this project, all the CC and Chance cards work as they would in the real game, moving the player occasionally, or adding or subtracting money, as well as distributing the Get Out of Jail Free cards. This was done to enhance the accuracy of the simulation. The decks are used up and then shuffled, just like in real play.

The data for n iterations were collected for the number of times landed on a space, and the corresponding percentage. The data is contained in the last section of this document.

Question Responses

The analysis of jail escape strategy A involved observing how the percentages of landing on each property changed with different values of n. As n increased, the percentages of landing on most properties stabilized. This indicated that higher values of n resulted in more consistent landing probabilities across the board.

Upon examining whether the probabilities converged between runs, it was observed that similar outcomes occurred across all iterations. The probabilities became stable, with percentages becoming very similar to each other across all ten iterations in the data concerning strategy A.

Further investigation sought to determine if there was a specific value of n that always produced nearly identical probabilities. It was found that certain squares were more likely to be landed on, such as jail. The jail square has multiple entry points, including the Go To Jail space, rolling triple doubles, or drawing a Community Chest or Chance card that sends a player to jail. This led to results showing that jail was about twice as likely to be landed on as any other individual property. The stabilization of results was more pronounced as n approached 1,000,000.

When comparing the two jail escape strategies (strategy A and strategy B) with n set to one million, the simulation revealed no significant differences in the overall probabilities of landing on most spaces. This suggested that the choice between strategies A and B did not materially impact the likelihood of landing on various board spaces.

The simulation was initially implemented for a single player. Considering whether the answers to the previous questions would change if the number of players increased, it was hypothesized that the results would remain largely unaffected. This is because Monopoly contains few moves that allow one player to influence the position of another player on the board. The number of players does not alter the probabilities associated with rolling any given number on the dice, nor does it affect the chance of landing on specific squares. Thus, the simulation's outcomes are expected to remain consistent regardless of the number of players involved.